

ProBuilder Upgrade Kit

UpgradeKit is a tool that prepares your ProBuilder objects for upgrading to a new version of ProBuilder. This is necessary because ProBuilder requires your 3d models to have certain components in order to work, and the process of importing new versions of UnityPackage files will break those script references. This tool searches your project for ProBuilder components, copies their data to a new component, then re-builds the ProBuilder components.

Upgrading Overview

- Import [ProBuilderUpgradeKit.unitypackage](#) to your project.
- For each scene in your project, run [Tools / ProBuilder / Upgrade / Prepare Scene for Upgrade](#).
- Delete the [ProCore / ProBuilder](#) folder. Other folders inside the ProCore directory may remain.
- Import new ProBuilder version.
- For each scene in your project, run [Tools / ProBuilder / Upgrade / Re-attach ProBuilder Scripts](#).

Detailed Steps

Import [ProBuilderUpgradeKit.unitypackage](#)

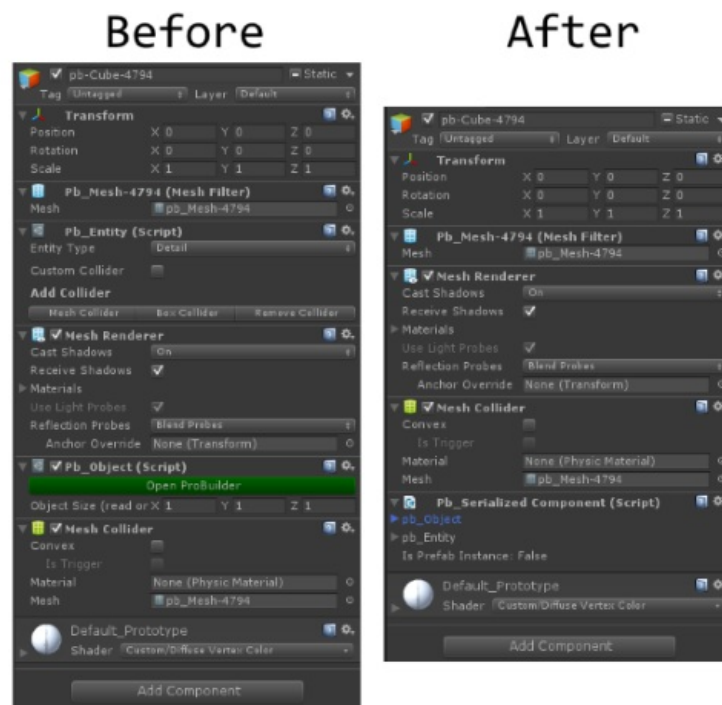
Open your project, and open the Console. Hit [Clear](#), to remove one-off warnings, and let the window reload. If there are persistent errors, fix them before continuing with the upgrade process. Compile errors will prevent ProBuilder scripts from executing properly, which may affect the upgrade process.

When your project is error-free, import [ProBuilderUpgradeKit.unitypackage](#).

Prepare Scenes for Upgrade

In each scene that contains ProBuilder objects, run the menu item [Tools / ProBuilder / Upgrade / Prepare Scene for Upgrade](#). You must run this command for every scene in your project that has any ProBuilder components present. Make sure to save after running this action (a dialog will prompt you to do so after completion).

You may see some warnings about prefab parents failing to serialize. This means the UpgradeKit has discovered prefabs that are present in your Project, but not in any scene. To properly serialize these objects, an instance is required in a scene.



Delete ProBuilder Folder

Prior to importing a new version of ProBuilder, delete the old ProBuilder folder. **You only need to delete the folder named "ProBuilder"** - the ProCore folder and other items in that folder may remain.

After this step there will be errors in the Console. This is expected.

Import New ProBuilder Package

Import the new ProBuilder version from the Asset Store or User Toolbox. Once the package is finished decompressing, the errors in the console should be absent.

Rebuild ProBuilder Components

In each scene that contains ProBuilder components, run [Tools / ProBuilder / Upgrade / Re-attach ProBuilder Scripts](#). The first time this menu item is run it will also rebuild prefab components.

If there are errors in the rebuilding phase, check the Console for more information. If an object failed to rebuild itself, the [pb_SerializableComponent](#) will not be removed, meaning your data is still safe.

Troubleshooting

Missing Script References

This means that this object was not serialized by the [Prepare Scene for Upgrade](#) action. Make sure you've run this menu item for the containing scene, and saved the scene prior to upgrading.

If you did not make a backup, it is possible to remove the missing components and run [Tools / ProBuilder / Actions / ProBuilderize](#) to make this object editable again.

After upgrade, the wireframe moves independently of the mesh

There are two [pb_Object](#) components on your GameObject. Remove one. Do the same for duplicate [pb_Entity](#) components. If there are many instances, running [Prepare Scene for Upgrade](#) and [Re-Connect ProBuilder Scripts](#) again in your scene will fix them en masse.